

Alberto Sánchez López
Antonio San Agustín García
Alberto Campos Blanco

Trabajo SALT

VARIABLE ALEATORIA

Lanzamiento de jabalina



EXPLICACIÓN DEL EXPERIMENTO

Hemos elegido una competición de **lanzamiento de jabalina** como un experimento aleatorio, cuya variable será el ángulo de lanzamiento de los atletas. El objetivo es ver como lanzadores con mayor altura y fuerza (velocidad de lanzamiento), factores muy importantes en el alcance de la jabalina, pueden quedar por detrás, en una clasificación, debido a que **el ángulo es una variable aleatoria gaussiana**. El objeto lanzado describe un **movimiento parabólico con cierta altura inicial**. Cada uno de los participantes tiene una serie de atributos: velocidad inicial que le imprime, altura y la dispersión angular.

El lanzador de jabalina, al iniciar su intervención, adquiere una energía cinética. La **velocidad del lanzamiento final** dependerá de la calidad de la transmisión de dicha energía a la extremidad superior, y de ésta a la jabalina. Según estudios realizados, la velocidad inicial abarca valores de entre 29 y 31 m/s (competidores mundialistas y olímpicos)

En cuanto a la **altura inicial**, estudios resaltan que lanzadores de nivel internacional lanzan desde 1,84 y 1,99 metros, y que no es el parámetro que marque las verdaderas diferencias en competición. Antes de lanzar, el atleta inclina el tronco superior (espalda, cadera, hombros, brazos) y las piernas hacia atrás, para imprimirle más fuerza, pero la altura final desde la que lanza es mayor a la altura propia. Alrededor de cuatro y seis centímetros mayor.

El **ángulo de lanzamiento** es un factor muy importante. Se le conoce también como el ángulo de ataque, resultado de la diferencia del ángulo de posición y el ángulo del vector velocidad en el instante de despegue. Normalmente, la jabalina sale disparada entre los 30 y los 36 grados. Depende de una secuencia de factores biomecánicos (inclinación y ángulo de rotación de hombros y caderas), tenidos en cuenta en la dispersión.

Definidos los tres factores diferenciales, se muestra a continuación una tabla para ver como varían en función de cada uno de los lanzadores y como los hemos escogido.

Lanzador	V inicial [m/s]	Altura [m]	Ángulo [°]	Dispersión
Vítězslav Veselý	29.65	1.91	34.32	4.1
Tero Pitkämäki	29.53	1.99	35.25	4.3
Dmitri Tarabin	29.68	1.88	34.15	4.4
Julius Yego	29.20	1.82	33.81	5
Roman Avramenko	29.32	1.90	33.91	4.5
Antti Ruuskanen	29.22	1.96	34.26	4.8
Andreas Thorkildsen	29.27	1.88	34.11	4.8
Ihab Abdelrahman	29.25	1.98	34.19	4.8

Partiendo de que **los lanzadores son los que compitieron en el último campeonato mundial**, nos hemos basado en los resultados de esta competición y en los de todo el año para elaborar una media de sus lanzamientos para poder adecuar las velocidades exactas y la dispersión. Así, los atletas con mayor dispersión, son aquellos con menos ángulo y velocidad inicial, para hacer más aleatorio el proceso. La **altura** de lanzamiento se ha realizado sabiendo la altura de cada uno de ellos y sumándole entre 4 y 6 centímetros. A los que clasificaron primero durante las pruebas en 2013, se les ha añadido mayor altura de lanzamiento. No es un factor que haga variar la distancia de una forma significativa, pero hay que tenerlo en cuenta ya que el lanzamiento no se realiza desde el suelo. Los **ángulos de lanzamiento** han sido escogidos en torno a los 34 grados que según estudios de lanzamiento de jabalina, es con el que lanzan los atletas internacionales.

El verdadero factor diferencial en el lanzamiento de jabalina, de entre los tres mencionados anteriormente, es el de la **velocidad de lanzamiento**. Es el que marca las diferencias. Sin embargo, hemos elegido el ángulo con el que sale disparada, porque abarca mayores valores que la velocidad inicial.

En cuanto al **funcionamiento de nuestro programa**, se simulará una hipotética competición 1000 veces, y se mostrará el porcentaje de victorias de cada participante. Una victoria es ser el lanzador que ha lanzado más lejos la jabalina. Posteriormente, se realiza la competición. Cada lanzador lanza una vez y se deciden las medallas. Cabe destacar, como en una prueba oficial, que puede haber un lanzamiento nulo. Se dice que un lanzamiento es nulo cuando el lanzador pisa o atraviesa la línea de lanzamiento antes, lanzando o después de lanzarla, cuando la jabalina sale fuera de las líneas del campo establecidas o no cae con la pica por delante, o cuando hay un retraso en la ejecución del lanzamiento. Nos hemos tomado una pequeña licencia, ya que

queríamos incluir esta posibilidad: hemos dado por lanzamiento nulo, aquel que no sobrepase los 60 metros de longitud. Debido a la aleatoriedad del suceso, nos salían valores de lanzamiento muy bajos. Y los hemos catalogado como nulos para dar más realismo al suceso.

ALEATORIEDAD DEL SUCESO

El método **nextGaussian()**, aplicado a un valor aleatorio (gracias a la función Random) definido anteriormente, asignado al ángulo y al escalamiento de la dispersión, hacen de este suceso, uno de tipo aleatorio. La explicación del código matemáticamente sería:

$$N(\text{ángulo}, \text{dispersión}) = \text{dispersión} * N(0, 1) + \text{ángulo}.$$

DESCRIPCIÓN DEL CÓDIGO JAVA EMPLEADO EN LA SIMULACIÓN

El código empleado para la simulación de los lanzamientos está estructurado en 4 clases; la clase **"Funcion"** que recoge todas las herramientas para el cálculo de los datos y además extiende las diversas clases **"LanzadorX"** que contiene a su vez los datos de cada uno de los lanzadores; la clase **"Lanzamiento"** encargada de inicializar todos los objetos (lanzadores) y lanzar los métodos que calculan los porcentajes, distancias y vencedores de la competición; y por último la clase **"PruebaLanzamiento"** que contiene el método Main (principal) de la simulación encargado de lanzar el programa, además de la representación y de los tiempos de ejecución.

A continuación explicaremos más detalladamente el código empleado en el programa:

- **Clase Funcion:** en esta clase definimos todo lo relacionado con las características físicas de los lanzadores, además de los métodos con las ecuaciones empleadas para el cálculo de la parábola del lanzamiento. El método *getAngulo()* es el encargado de calcular el ángulo de lanzamiento después de haber aplicado la dispersión con la gaussiana. Los métodos *getVLanzador()* y *getAltura()* nos permite conocer la velocidad inicial de lanzamiento y la altura de cada lanzador definida en cada una de las clases **"LanzadorX"** para cada uno de los participantes. Y por último el método *medirAlcance()*. Es el método más importante de esta clase. Se crean dos variables de tiempo. Partiendo de las ecuaciones del movimiento parabólico, despejamos el tiempo y llegamos a una ecuación de segundo grado. El problema es que una de las soluciones es negativa, por eso hemos creado dos variables. Y con un condicional, conseguimos así solo la que nos interesa.

Por último queda que devolver el alcance: velocidad de inicio (característica de cada lanzador) multiplicado por el coseno del ángulo de lanzamiento y por el tiempo.

- Clase LanzadorX: contiene los datos de cada lanzador extendiendo la clase anterior. Cada lanzador consta de velocidad inicial de lanzamiento (m/s), estatura (m), grado de dispersión (grados), y ángulo de lanzamiento (grados).
- Clase Lanzamiento: en esta clase se inicializan todos los objetos (lanzadores). El método *Redondear()* lo utilizamos para redondear con dos decimales los resultados. El método *getAlcances()* se encarga de imprimir por pantalla todos los lanzadores de la competición con sus respectivos resultados obtenidos aleatoriamente en una tirada. El método *getVencedores()* *junto con el método anterior se encarga de imprimir por pantalla los ganadores de la competición. Y por último el método lanzar1000()* se encarga de simular 1000 lanzamientos e imprimir por pantalla el porcentaje de victorias de cada uno de los lanzadores.
- Clase PruebaLanzamiento: que contiene únicamente el método Main encargado de reproducir por orden y respetando los tiempos de ejecución, la probabilidad de victoria de cada uno de los lanzadores, los alcances de cada uno de los lanzadores, y el vencedor de estos.

```
/**
 * Despejando en las ecuaciones del movimiento parabolico el alcance con el dato de altura inicial,
 * medimos la distancia de lanzamiento
 * @param vLanzador Velocidad inicial del lanzador
 * @param alturaLanzador Altura del lanzador
 * @return alcance de la jabalina
 */
public double medirAlcance(double vLanzador, double alturaLanzador){
    double g = 9.80665;//Gravedad
    double t=0;
    //Tomando el signo positivo
    double t1 = (-vLanzador*Math.sin(getAngulo()) +
        Math.sqrt( vLanzador*vLanzador*Math.sin(getAngulo())*Math.sin(getAngulo()) + 2*g*alturaLanzador))/
        (-g);
    //Tomando el signo negativo
    double t2 = (-vLanzador*Math.sin(getAngulo()) -
        Math.sqrt( vLanzador*vLanzador*Math.sin(getAngulo())*Math.sin(getAngulo()) + 2*g*alturaLanzador))/
        (-g);
    //Con este condicional elegimos la de signo positivo si cumple la condicion
    //O la de signo negativo si la anterior no la cumple
    if(t1>0){ t=t1; }
    else{ t=t2; }

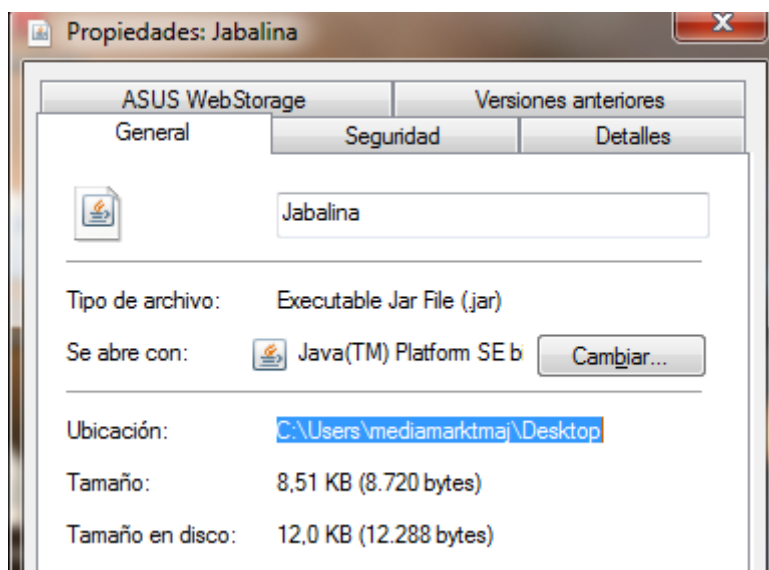
    return vLanzador*Math.cos(getAngulo())*t;
}
```

Código del método medirAlcance()

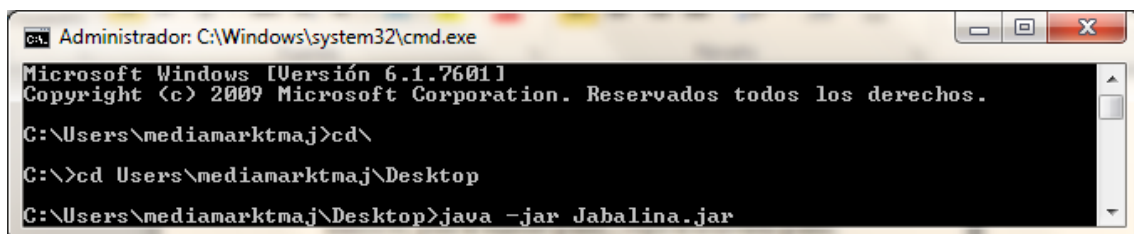
¿CÓMO EJECUTAR EL PROGRAMA?

Para la ejecución del programa usaremos la consola de Windows (cmd) o el terminal de Linux. A continuación explicaré brevemente como ejecutarlo desde la consola de Windows (similar en Linux).

1. Abrimos la consola y escribimos en ella: “**cd/**” (con esto accedemos al directorio principal del disco duro)
2. A continuación buscamos el archivo Jabalina.jar adjunto en el trabajo y en propiedades obtenemos la ruta del archivo. En mi caso “**C:\Users\mediamarktmaj\Desktop**”.



3. Escribimos en la consola la dirección anterior: “**cd Users\mediamarktmaj\Desktop**” (quedando indicada en la consola la ruta del archivo).
4. Y por último ejecutamos el archivo.jar escribiendo: “**java -jar Jabalina.jar**”



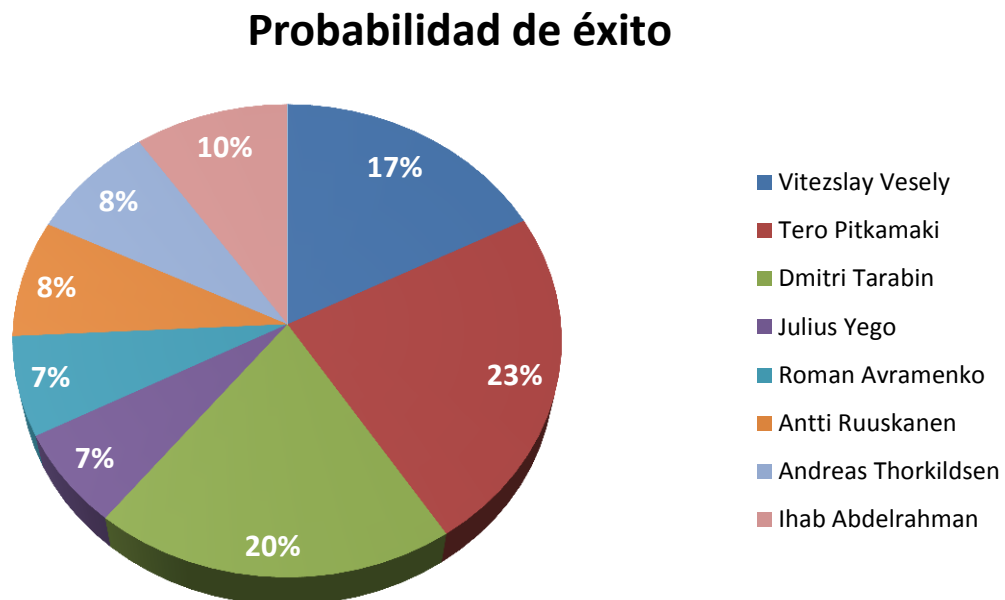
EJEMPLO PRÁCTICO DE LA SIMULACIÓN

Una vez ejecutado el programa, tal y como hemos explicado anteriormente, aparecen las probabilidades de victoria de cada lanzador, producto de lanzar mil competencias y calcular el porcentaje de victorias.

```
En esta competición participan 8 lanzadores de los que
centaje de éxito:

Vitezslay Vesely: 17.5% <opciones de ganar>
Tero Pitkamaki: 23.3% <opciones de ganar>
Dmitri Tarabin: 20.0% <opciones de ganar>
Julius Yego: 6.6% <opciones de ganar>
Roman Avramenko: 6.8% <opciones de ganar>
Antti Ruuskanen: 8.0% <opciones de ganar>
Andreas Thorkildsen: 8.1% <opciones de ganar>
Ihab Abdelrahman El Sayed: 9.7% <opciones de ganar>
```

Con los datos anteriores obtenemos la siguiente gráfica donde se puede ver con claridad cuál de los 8 lanzadores tiene más opciones de ganar:

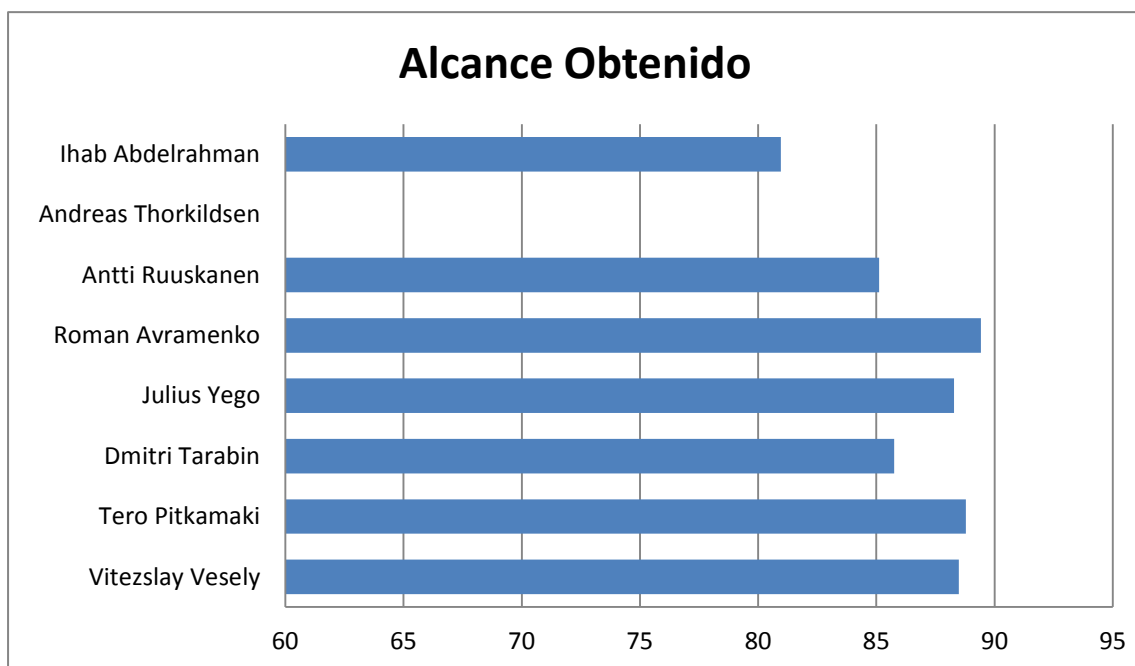


A continuación, transcurrido el tiempo de espera para el inicio, se ejecuta un único lanzamiento por cada jugador.

```
La competicion de lanzamiento de jabalina va a comenzar
3, 2, 1 ...

Vitezslay Vesely: 88.49 m
Tero Pitkamaki: 88.78 m
Dmitri Tarabin: 85.75 m
Julius Yego: 88.29 m
Roman Avramenko: 89.42 m
Antti Ruuskanen: 85.11 m
Andreas Thorkildsen: Nulo
Ihab Abdelrahman El Sayed: 80.96 m
```

En la siguiente gráfica podemos observar con mayor claridad quien ha sido el vencedor, en este caso el ganador ha sido “Roman Avramenko”. También vemos que “Andreas Thorkildsen” ha obtenido un alcance nulo, esto se debe a que hemos dado por nulos los lanzamientos con alcance menor de 60 metros.



Y por último, elegimos los 3 primeros vencedores según sus resultados.

```
Vitezslay Vesely ha conseguido la medalla de bronce con una distancia de: 88.49 m
Tero Pitkamaki ha conseguido la medalla de plata con una distancia de: 88.78 m
Roman Avramenko ha conseguido la medalla de oro con una distancia de: 89.42 m
```